PERGAMON

Pattern Recognition 35 (2002) 927–944

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

Optimal camera placement for accurate reconstruction

Gustavo Olague[a,*], Roger Mohr[b]

[a] *Departamento de Ciencias de la Computación División de Física Aplicada, Centro de Investigación Científica y de Educación Superior de Ensenada, B.C. Km. 107 carretera Tijuana-Ensenada, 22860, Ensenada, BC, Mexico*
[b] *MOVI-GRAVIR[1], ZIRST-655 Avenue de l'Europe, 38330 Montbonnot Saint Martin, France*

Received 17 March 2000; accepted 2 April 2001

Abstract

Three-dimensional (3D) measurements can be recovered from several views by triangulation. This paper deals with the problem of where to place the cameras in order to obtain a minimal error in the 3D measurements, also called *camera network design* in photogrammetry. We pose the problem in terms of an optimization design, dividing it into two main components: (1) an analytical part dedicated to the analysis of error propagation from which a criterion is derived, and (2) a global optimization process to minimize this criterion. In this way, the approach consists of an uncertainty analysis applied to the reconstruction process from which a covariance matrix is computed. This matrix represents the uncertainty of the detection from which the criterion is derived. Moreover, the optimization has discontinuities due to the presence of occluding surfaces between the viewpoint and the object point group, which leads to a combinatorial optimization process. These aspects are solved using a multi-cellular genetic algorithm. Experimental results are provided to illustrate the effectiveness and efficiency of the solution. © 2002 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved.

Keywords: Camera network design; Uncertainty analysis; Global optimization; Covariance matrix; Genetic algorithms

1. Introduction

Visual measurements can be made significantly more accurate by selecting a well designed convergent camera configuration and a suitable mathematical model of the process. As an example, see the work of Fraser [1] in which he reports highly accurate measurements employing a strong (hand-designed) photogrammetric network using 36 photographs taken by 18 well disposed cameras with high convergence angles. Another example is the work of Beyer [2] in which he reports the accuracies of 1/100th of a pixel using CCD cameras. Despite achievements like these and due to the complexity of designing the convergent multi-station networks, the photogrammetric measurement technique has rarely been applied by those other than the experienced photogrammetrists.

The problem of automating the camera network design process in order to attain highly accurate measurements has received comparatively little attention given its practical importance. This paper deals with the problem of determining the positioning of several cameras observing a set of targets placed on a 3D object with the goal of achieving highly accurate measurements. The input data would be the approximately known set of target points to be measured in order to obtain the best accuracy in the final measurements. Camera network design is hard to obtain due to all the unknown number of configurations having a very similar accuracy, but with a very different imaging geometry. Several additional constraints like the upper bound on the number of cameras, and placement limitation due, for example, to the incidence

* Corresponding author. Tel.: +52-6-174-5050 ext 25402; fax: +52-6-175-0549.
E-mail addresses: golague@cicese.mx (G. Olague), roger.mohr@inrialpes.fr (R. Mohr).

[1] MOVI is a joint research project between the CNRS, INPG, INRIA Rhône-Alpes and UJF.

0031-3203/02/$22.00 © 2002 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved.
PII: S0031-3203(01)00076-0

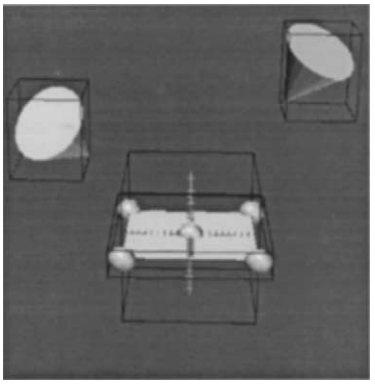


Fig. 1. Two convergent cameras. The cones illustrate the camera attitudes and the ellipsoids represent the uncertainty of the observed targets.

angle must be taken into account in the exploration of the search space. Automating such a camera placement process reaches a high complexity due to the competing nature of the constraints and numerous design decisions that need to be made. The mathematical model of triangulation plays a key role as it leads to the uncertainty model and therefore to the error criterion to be optimized. The paper presents a prototype system called EPOCA (an acronym for "evolving positions of cameras") that we have implemented to test the validity of our approach.

This paper is organized as follows: Section 1 discusses the problem, isolating several key issues, and reviews the literature related to sensor planning. Later, we describe our approach in one main section. Section 2 is divided into five subsections; three of them (the reconstruction process, 3D-error estimation and image error estimation) are used in the development of a criterion. Then, our method of camera placement is presented in Section 2.5. It is implemented as a multi-cellular genetic algorithm in order to optimize the criterion. Finally, we present some experimental results in Section 3 followed by a discussion in Section 4.

1.1. Statement of the problem

The problem we approach is the automation of the camera network design process in order to obtain accurate 3D measurements. We restrict ourselves to the problem where the cameras remain at a fixed distance to the set of target points to be measured. The placement is limited due to self-occlusion produced by the object shape. Imagine that we would like to accurately measure some fiducial marks distributed over a planar surface, and that we would like to place the cameras to obtain a minimal error. Fig. 1 shows two cameras observing five target points, represented by error ellipsoids describing the uncertainty of their position. Changing the camera attitudes (position and orientation) changes the orientation and size of the ellipsoids. The main question we would like to answer is presented in Table 1. From these initial questions the choice of a *criterion* combined with an *optimization* process will be the key decisions.

Table 1
Some initial questions

- Where should we place the cameras in order to obtain the minimal 3D error?

From this question several subproblems arise:

1. How can we develop a good criterion to judge our configuration?
2. What conditions are needed for our system to work?
3. Which are the interrelated aspects involved in the development of the system?
4. What would be a good method to optimize the placement of the camera?

1.2. Related work

Researchers in the computer vision and photogrammetric communities have recognized the need to automate the process of camera network design. The literature devoted to this problem, however, is limited due to various complex aspects that are involved. This section surveys the existing literature on automated single and multiple camera placements.

Review of single camera systems: The HEAVEN system developed by Sakane et al. [3] is an example of a system that finds possible sensor positions using a *generate-and-test strategy*, for the inspection of an object tessellated by a sphere of a given radius. The object is not occluded but an analysis of light source placement is incorporated. This study was extended to the VIO system developed by Sakane and Niepold [4]. Another example of the generate-and-test approach for camera and light positions is ICE [5]. This places emphasis on the illumination-planning component of the problem. For camera placement, the main task constraint considered is edge visibility. The sensor is positioned to minimize the occlusion of selected feature edges. The evaluation of this criterion is based on an aspect graph representation of the object. Other works have adopted a *synthesis approach*. Instead of taking a discrete approach, the idea is to model constraints as analytical functions. *The automatic sensor and illuminator positioning work of SRI* carried out by Cowan et al. [6] optimizes the camera locations from which a specified set of object features can be viewed. Each of these view-points satisfies a general set of geometric constraints, i.e., all features appear within the field-of-view, in focus, at sufficient resolution and incidence angle, and unoccluded by the object itself or other objects in the working environment. The MVP system developed by Tarabanis et al. [7] determines the optimal sensor location and sensor parameters (focal length,

focus setting and aperture) for viewing a set of surfaces and avoiding occlusion. MVP is currently being extended to work in dynamic environments [8]. The placement of illumination sources for inspecting edge features has also been investigated by Cowan et al. [9]. Another interesting problem is known as the next best view [10]. It is involved in determining the areas of a scanner's viewing volume that need to be scanned to sample all of the visible surfaces of an a priori unknown object and in the positioning of the scanner. While all these systems provide solutions to the task of viewing an object, they do not address the task of deriving accurate object dimensions.

Review of multiple camera systems: Tarbox and Gottschlich [11] have recognized the need for *multi-station solutions* to overcome object occlusion problems. They have implemented a solution in the IVIS system for an active triangulation sensor. Cowan et al. [12] have also experimented methods to place multiple sensors overcoming the occlusion problem associated with 3D objects. Fritsch and Crosilla [13] have investigated the potential of optimizing multi-station configurations using an analytical first order design (FOD) approach by iteratively shifting the sensor stations until the covariance matrix of the estimated object feature coordinates was better than a criterion matrix. However, their approach is limited to the optimization of existing configurations. The user, prior to the optimization, makes critical decisions, for example, the number of camera stations and their positioning for a good approximation to a strong configuration. Photogrammetrists affirm that an infinite number of possible camera station poses exists just limited by the set of competing constraints. The complexity of camera placement belongs to the class of NP-complete[2] problems, see Ref. [14]. Consequently, it can be concluded that a trial and error approach to FOD is unsatisfactory and not tractable by traditional methods. Mason and Grün [15] developed a work called CONSENS that follows the expert system approach and uses multiple cameras in combination with optical triangulation. It outlines a method of overcoming these limitations. The method is based on the theory of *generic networks*, which constitutes compiled expertise, describing an ideal configuration of four camera stations that can be employed to provide a strong imaging geometry for the class of planar network design problems. Complex objects are divided into planes; each one is evaluated through one of these networks and then connected with additional cameras with the purpose of establishing just one common datum.

Unlike all other approaches, our idea is to pose the problem in terms of a global optimization design, which is capable of managing the problem using an adaptive strategy. It explores the solution space using both non-continuous optimization and combinatorial search.

[2] All known exact algorithms for NP-complete problems run, in the worst case, in a time exponential in the size of the input data. It is widely conjectured that this will not change in the future.

2. The approach

This section presents a method to approach the problem of optimal camera placement in order to obtain accurate positions of 3D target points [16]. This approach can be divided into two main components. We will first develop an analytic uncertainty analysis based on the error propagation process. This will allow us to express an error criterion to be minimized. Second, we will present an evolutionary optimization method similar to *genetic algorithms* (see Ref. [17]), which optimizes this criterion.

2.1. Three-dimensional reconstruction

Let u_{ij} and v_{ij} denote the photo coordinates of the image of point j in photograph i. For each pair of image coordinates $(u_{ij}, v_{ij})^{\mathrm{t}}$ observed on each image, the following relationship exists:

$$u_{ij} = \frac{m^i_{11}X_j + m^i_{12}Y_j + m^i_{13}Z_j + m^i_{14}}{m^i_{31}X_j + m^i_{32}Y_j + m^i_{33}Z_j + m^i_{34}},$$

$$v_{ij} = \frac{m^i_{21}X_j + m^i_{22}Y_j + m^i_{23}Z_j + m^i_{24}}{m^i_{31}X_j + m^i_{32}Y_j + m^i_{33}Z_j + m^i_{34}}. \quad (1)$$

This system of equations assumes that light rays travel in straight lines, that all rays entering a camera lens system pass through a single point and that the lens system is distortionless or, as is usual in a highly accurate measurement, that the distortion has been cancelled out after having been estimated. The imaging process just described is known as the pinhole camera model. This model is based on the fundamental assumption that the exposure center, the ground point, and its corresponding image point, all lie on a straight line. In this way, a point in the scene P_j, $j = 1, \ldots, n$, of homogeneous coordinates $(X_j, Y_j, Z_j, 1)^{\mathrm{t}}$ is projected into points p_{ij} of image coordinates $(u_{ij}, v_{ij})^{\mathrm{t}}$, through a projection matrix M_i, $i = 1, \ldots, m$, of size 3×4 corresponding to the ith image. Therefore, three-dimensional measurements can be obtained from several images. Each matrix M represents a mapping composed of a transformation $W \rightarrow C$ from the world coordinates W to the camera coordinates C given by

$$\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{bmatrix} R_{WC} & T_{WC} \\ 0_{1\times 3} & 1 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}, \quad (2)$$

where the rotation matrix R_{WC}, which is a function of three rotation parameters (α, β, γ) and the translation vector T_{WC}, also of three degrees of freedom, characterizes the camera's orientation and position with respect to the world coordinate frame. Under the perspective projection, the transformation from 3D-world coordinate system to the 2D-image coordinate is

$$\begin{pmatrix} su \\ sv \\ s \end{pmatrix} = K \begin{bmatrix} R_{WC} & T_{WC} \\ 0_{1\times 3} & 1 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}, \tag{3}$$

where the matrix

$$K = \begin{pmatrix} k_u f & 0 & u_0 & 0 \\ 0 & k_v f & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix},$$

represents the intrinsic parameters of the camera; f is the focal length of the camera, (k_u, k_v) the horizontal and vertical pixel sizes on the image plane, and (u_0, v_0) the projection of the camera's center (principal point) on the image plane. So, a camera can be considered as a system that performs a known linear projective transformation from the projective space $\mathbb{P}^3$ into the projective plane $\mathbb{P}^2$. Considering the projective matrix of each camera and the image points as known, and that each image point is projected by one and only one projection matrix, we can develop an error propagation study using Eq. (1) that can be rewritten as follows:

$$\begin{bmatrix} (u_{ij}m^i_{31} - m^i_{11})\ (u_{ij}m^i_{32} - m^i_{12})\ (u_{ij}m^i_{33} - m^i_{13}) \\ (v_{ij}m^i_{31} - m^i_{21})\ (v_{ij}m^i_{32} - m^i_{22})\ (v_{ij}m^i_{33} - m^i_{23}) \end{bmatrix}$$

$$\times \begin{pmatrix} X_j \\ Y_j \\ Z_j \end{pmatrix} = \begin{bmatrix} (m^i_{14} - u_{ij}m^i_{34}) \\ (m^i_{24} - v_{ij}m^i_{34}) \end{bmatrix} \tag{4}$$

or in matrix notation, considering one object point,

$$A(p_i, M_i)P = b(p_i, M_i), \tag{5}$$

where A is a $2i \times 3$matrix and b is a $2i \times 1$ vector. Since the inverse of A^tA can be computed, we can find the least squares solution

$$P = (A^tA)^{-1}A^tb, \tag{6}$$

which minimizes $\| AP - b \|^2$.

This equation expresses the linear relationship between the image points p_{ij} and the three-dimensional points P_j through the projective matrices M_i. The problem involves a set of cameras viewing P points placed over S surfaces. We would like to know the best camera network in order to reduce the uncertainty of the reconstructed points. The method of solving the system is similar to that proposed by several photogrammetrists [18,19], known as *limiting error propagation*. It makes the assumption that the projective parameters are error free and that the variances in the object point coordinates arise solely from the propagation of random errors in the image coordinates measurements. This method has proven to be effective in the case of strong networks. However, a definition of a favourable configuration of cameras in position and orientation in order to produce a strong configuration must be given. This is exactly what our system provides, as we shall observe in the experiments.

2.2. 3D error estimation

Until now, we have studied the function to transform a set of image points into a point in space

$$P = f(p), \tag{7}$$

given by Eq. (6), which is going to be useful in developing an analysis of error propagation, see Ref. [20]. The key to manipulating geometric uncertainty is to be able to transform the information or probability density function on a feature available in one form (image point) into another form of interest (point in space). This transformation of information can be grouped into a family of transformations that we approximate to the exact transformation by a first-order relation using a Taylor series. Successive moments can be found by equating higher-order terms; however, using higher-order terms is neither viable nor desirable, as any computational simplicity would be lost. In this way, a linear approximation is to be used in which we assume a Gaussian distribution. Then, the mean $E[P]$ and covariance ΛP are sufficient information to completely define the feature density function. This is given by the following proposition [20, Chapter 5]:

Proposition 1. *Given a random variable* $p \in \mathbb{R}^m$, *of Gaussian distribution, mean* $E[p]$, *and covariance* Λp, *and* $P \in \mathbb{R}^n$, *the random vector given by* $P = f(p)$, *where* f *is a function of class* C^1, *the mean of* P *can be approximated to a first-order Taylor expansion by* $f(E[p])$ *and its covariance by*

$$\Lambda P = \frac{\partial f(E[p])}{\partial p}\, \Lambda p\, \frac{\partial f(E[p])^t}{\partial p}. \tag{8}$$

Proof. A first-order Taylor expansion of f in the vicinity of $E[p]$ yields

$$f(p) = f(E[p]) + \frac{\partial f(E[p])}{\partial p}\,(p - E[p]) + \Theta(p). \tag{9}$$

The term $\partial f(E[p])/\partial p$, represents a Jacobian matrix of $f(p)$ in $E[p]$ and $\Theta(p)$ is a bounded function in each coordinate given by $\varepsilon(\| p - E[p] \|^2)$, where the function $\varepsilon : t \to \varepsilon(t)$ from $\mathbb{R}$ into $\mathbb{R}$ is such that $\lim_{t\to 0} \varepsilon(t) = 0$. By

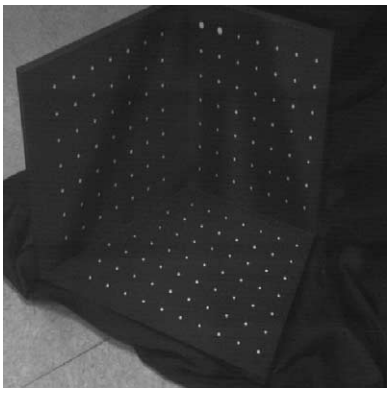


Fig. 2. The calibration grid. This grid is composed of retro-reflective targets.

assuming now that any sample of p is sufficiently close to $E[p]$, we can approximate f to first-order by using Eq. (9), which yields

$$E[P] \simeq f(E[p]),$$

$$f(p) - f(E[p]) \simeq \frac{\partial f(E[p])}{\partial p}(p - E[p]).$$

We then have

$$E[(f(p) - f(E[p]))(f(p) - f(E[p]))^{t}]$$

$$\simeq E\left[\frac{\partial f(E[p])}{\partial p}(p-E[p])(p-E[p])^{t}\left(\frac{\partial f(E[p])}{\partial p}\right)^{t}\right]$$

$$= \frac{\partial f(E[p])}{\partial p} E[(p-E[p])(p-E[p])^{t}]\left(\frac{\partial f(E[p])}{\partial p}\right)^{t},$$

which gives us a first-order approximation of the covariance matrix of P as a function of the covariance matrix of p given by Eq. (8).

Therefore, ΛP is a symmetric positive definite matrix, which describes the bounds on $P = f(p)$ in the vicinity of $E[P] = f(E[p])$ given by those of p in the vicinity of $E[p]$. This proposition lets us compute the uncertainty of the three-dimensional point knowing the uncertainty in the image points. Another model is needed to give an interpretation of matrix Λp, which describes bounds on the possible values of the coordinates of p. This implies the need for an image error model.

2.3. Image error estimation

In order to complete the uncertainty analysis, a study concerning the image points must be done. In this way, experiments can be carried out to detect the circular targets of the calibration grid (see Fig. 2). This grid is composed of 160 retro-reflective circular targets within a volume of $33.7 \times 26.2 \times 26.2\,\mathrm{cm}^3$. Images have been acquired with a PULNIX TM-6EX camera and a KINOPTIK lens of focal length 12.5 mm. The frame grabber is an "Imaging Technology 150". As a means to estimate the covariance matrix Λp of our 2D measured points, we will use Proposition 1. Therefore, we must have a function to relate image point errors taken from several photographs over different angles. The relation will be established using the cross-ratio, see Ref. [21]:

$$f(k) = k(a,b,c,d) = \frac{(c-a)(d-b)}{(d-a)(c-b)}, \tag{10}$$

which is a projective invariant. It lets us compute the uncertainty of the cross-ratio as a function of the uncertainty of the image points. In the case of homographic configurations of four points, the uncertainty varies in a way inversely proportional to the distances between the points, see Ref. [22]. In this way, considering the uncertainty of the image points as identical, we can select a configuration of four points, $f(k) = [0,3,2,1] = 4$, using the retro-reflective targets of the calibration grid in order to produce stable configurations. The cross-ratio can be linearized locally with respect to configuration $f(k) = [0,3,2,1]$. The relationship between the standard deviation of the cross-ratio and the uncertainty of the image points can be obtained by a first-order Taylor expansion of Eq. (10):

$$\Lambda k = \partial f(k) \Lambda p \, \partial f(k)^{t}, \tag{11}$$

where Λk of size 1×1, is equal to the cross-ratio variance σ_k^2. This value is computed considering all the cross-ratios, $f(k_i)$, of four aligned and equidistant targets, using the statistical method

$$\sigma_k^2 = \frac{1}{N-1}\sum_{i=1}^{N}(f(k_i) - \bar{f}(k))^2.$$

$\partial f(k)$ is the Jacobian of $f(k)$, where the partial derivatives with respect to each point are computed as follows:

$$\boldsymbol{J}_k = \left[\frac{(b-d)(c-d)}{(a-d)^2(b-c)}, \frac{(a-c)(d-c)}{(a-d)(b-c)^2}, \right.$$

$$\left. \frac{(b-d)(b-a)}{(a-d)(b-c)^2}, \frac{(a-c)(b-a)}{(a-d)^2(b-c)}\right].$$

Considering that under a homographic transformation

$$\boldsymbol{J}_k(\lambda a, \lambda b, \lambda c, \lambda d) = \frac{1}{\lambda}\boldsymbol{J}_k(a,b,c,d),$$

we obtain $\| \boldsymbol{J}_k(0,3,2,1) \|^2 = 80$. Λp is the covariance matrix of the four aligned points used to compute the cross-ratio $f(k)$. Furthermore, we can consider that all targets are identified without systematic errors,

$$\Lambda p = \sigma_p^2 \boldsymbol{I}_{4\times 4},$$

where σ_p represents the standard deviation of the target extraction, and $\boldsymbol{I}$ represents the unit matrix. In this way,

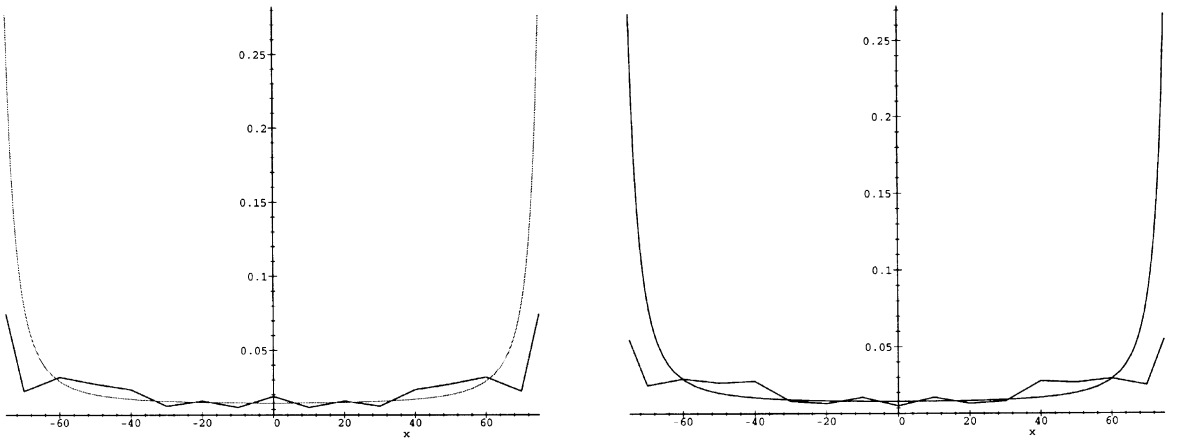


Fig. 3. The graphs show the best fit curve along both the axes. The X-axis corresponds to the incidence angle constraint (in degrees) and the Y-axis to the image error; (a) Best fit along x-axis direction. (b) Best fit along y-axis direction.

Eq. (11), can be rewritten as

$$\sigma_k^2 = \sigma_p^2 \parallel \boldsymbol{J}_k \parallel^2 \rightarrow \sigma_p = \sigma_k \frac{\lambda}{\parallel \boldsymbol{J}_k \parallel^2}, \tag{12}$$

where σ_k represents the standard deviation of the cross-ratio, λ the average distance between the targets, and $\boldsymbol{J}_k$ the Jacobian of $f(k)$. We assume that the image errors are Gaussian and identical (similar targets) along their alignment direction, and constant during the experiment (similar distance of observation and viewing angle). We use a subpixel target detector, on the calibration grid of Fig. 2, similar to one presented by Gruen [23]. From 100 retro-measures of four approximately equidistant points along each direction, we compute the uncertainty in the image points σ_p from the measured uncertainty of the cross-ratio σ_k, using Eq. (12). σ_k is a function of the observation angle α along their alignment direction. As the targets are circular and viewed as an ellipse, it can be assumed that the covariance of its two-dimensional location can be considered as a diagonal matrix. The graphs in Fig. 3 show final results of our experiments along the x and y directions. Consequently, the reference frame of the ellipse represented by two axes has each one-associate variance corresponding to the viewing angle in the two corresponding directions. These results can be approximated to a model using the Levenberg–Marquardt method [24], and the merit function [3]

$$y = \rho(\mathrm{e}^{(v/(90-x))} + \mathrm{e}^{(v/(90+x))} + \vartheta, \tag{13}$$

yielding the best fit parameters: $v = 79.74$, $\rho = 1.31 \times 10^{-3}$ and $\vartheta = 8 \times 10^{-3}$, for both experiments. This model corresponds to the incidence angle constraint in a camera network design. Note that for an angle larger than 80°, we have no more measurements, and approaching this value leads to an infinite uncertainty. The Λp model is useful to compute the covariance matrix ΛP of the 3D points.

2.4. The criteria

Once we have computed the covariance matrix ΛP, it is necessary to choose a criterion useful to the optimization process. In this way, we need to select a metric to compare symmetric positive definite matrices. The *comparison* of covariance matrices is interpreted as the required standard deviation of the function $P = f(p)$ to be better when it is computed with the covariance matrix ΛP_1 than with ΛP_2:

$$\Lambda P_1 \leqslant \Lambda P_2, \quad \sigma_f^{\Lambda P_1} \leqslant \sigma_f^{\Lambda P_2}. \tag{14}$$

In this way, a useful measure is the maximum eigenvalue λ_{max}, which can be computed from

$$\Lambda P q = \lambda q. \tag{15}$$

Moreover, the square root $\sqrt{\lambda_{max}}$ is related to the maximum standard deviation.

Another metric for comparing the covariance matrices ΛP and ΛQ (an ideal matrix) can be achieved by using the sum of the squared logarithms of the eigenvalues [25]:

$$d(\Lambda P, \Lambda Q) = \sqrt{\sum_{i=1}^{3} \ln^2 \lambda_i(\Lambda P, \Lambda Q)}. \tag{16}$$

[3] Note that these observations are better approximated to a hyperbolic function.

However, this metric requires a criterion matrix and due to the methodology, we cannot establish an ideal matrix, because we do not know the answer in advance. In fact, the maximum eigenvalue of ΛP may be replaced by a less tight norm in order to avoid the rigorous determination of the maximum eigenvalue, e.g. the trace

$$tr(\Lambda P) = \sum_{i=1}^{3} \Lambda P_{ii} = \sum_{i=1}^{3} \lambda_i(\Lambda P). \tag{17}$$

Since the maximum eigenvalue is related to the maximum standard deviation instead of the trace of ΛP, we propose to use as a measure $\mu(P)$, the *maximum element in the diagonal of* ΛP, which corresponds to the worst variance among the individual components, that is

$$\mu(P) = \max_{j=1,\ldots,3} \Lambda P_{jj}. \tag{18}$$

This measure is easy to compute and gives a good uniformity of network precision in each coordinate axis of the error ellipsoid represented by ΛP. The hyperellipsoid of uncertainty is not a criterion by itself, but it is useful to visualize the uncertainty of the three-dimensional measurements. If we define the random vector χ by

$$\chi = \Lambda P^{-1/2}(P - E[P])$$

and consider that P follows a Gaussian distribution, then χ follows a Gaussian distribution of zero mean and of covariance:

$$E[\chi\chi^{\mathrm{t}}] = E[\Lambda P^{-1/2}(P - E[P])(P - E[P])^{\mathrm{t}}\Lambda P^{-1/2}]$$
$$= \Lambda P^{-1/2}\Lambda P\Lambda P^{-1/2} = \boldsymbol{I}.$$

Consequently, the random variable δP, defined by

$$\delta P = \chi^{\mathrm{t}}\chi = (P - E[P])^{\mathrm{t}}\Lambda P^{-1}(P - E[P])$$

follows a χ^2 distribution of $t = rang(\Lambda P)$ degrees of freedom. Given a scalar s, and knowing the probability to be equal to $P_{\chi^2}(s,t)$, where δP appears between 0 and s; we have the following proposition:

Proposition 2. *If we consider that P follows a Gaussian distribution, the probability that P lies inside the s-hyperellipsoid defined by the equation*

$$(P - E[P])^{\mathrm{t}}\Lambda P^{-1}(P - E[P]) = s^2 \tag{19}$$

is equal to $P_{\chi^2}(s,t)$, *where s is any scalar, and t is the rank of* ΛP.

The s-hyperellipsoid makes it possible to represent graphically the uncertainty related to ΛP. Obviously, it can be done just in the case of $t \leqslant 3$. For a given t, we can compute the half-axes of the hyperellipsoid by

$$a_i = \sqrt{\frac{s^2}{w_i}}, \tag{20}$$

where w_i are the eigenvalues of the ΛP^{-1} matrix, and the directions of main axes $\boldsymbol{w}_i$ correspond to the eigenvectors.

2.5. The optimization process

The problem of camera network design presents discontinuities mainly due to the occlusion of targets, leading to a combinatorial optimization process, which we have approached using a multi-cellular genetic algorithm. Genetic algorithms are probabilistic parallel search techniques based on the mechanism of natural selection and natural genetics [17]. Since their development in the late 1960s [26], genetic algorithms have been proven effective in searching large, non-linear, complex and poorly understood search spaces, where expert knowledge is scarce or difficult to encode and where traditional optimization techniques fail. Automation of camera network design with the goal of achieving high accurate measurements is not direct. Many decisions need to be taken with the purpose of proposing one optimal configuration. The complexity of the problem becomes evident when we study complex objects using multiple cameras. Deterministic methods do not adapt very well. This is mainly due to the high number of design decisions in the form of thresholds. The system must take into account all the constraints in order to solve the numerical and combinatorial problem. In this way, genetic algorithms provide a general framework that is useful in solving this task. Moreover, it is necessary to note that network design is based on very precise rules such as imaging geometry and incidence angle. This allows us to differentiate a good configuration from a bad one. Genetic algorithm strategy is based only in the direct comparison of solutions. The set of solutions corresponding to the local minima does not have a significant difference with respect to the global minimum as photogrammetrists affirm, see Ref. [27]. This set of solutions, called here as *alternative solutions*, provides similar characteristics about the homogeneity of the ellipsoid of uncertainty. Consequently, we can conclude that they are of the same nature. However, all these configurations can be very different with respect to the imaging geometry. Stochastic methods do not have these problems. This is the main reason for selecting them. Therefore, a process of combinatorial search like genetic algorithms must find the different topologies that a set of cameras produces with respect to an object, taking into account the constraints that limit the search space. The knowledge of these configurations is an important step towards camera network design [28].

2.5.1. The adaptive system

The camera network design can be achieved following genetic algorithm methodology. This methodology is composed of five major components, see Fig. 4:

1. The definition of a structure, $A \in \alpha$, which represents a tentative solution to the problem. We represent it here as a set of variables, which are grouped into just one common structure, see Fig. 5.

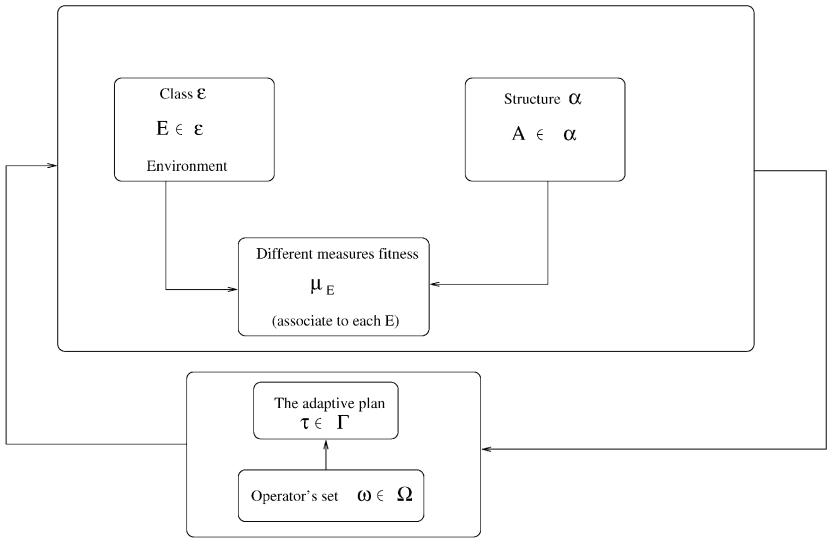


Fig. 4. The adaptive system.

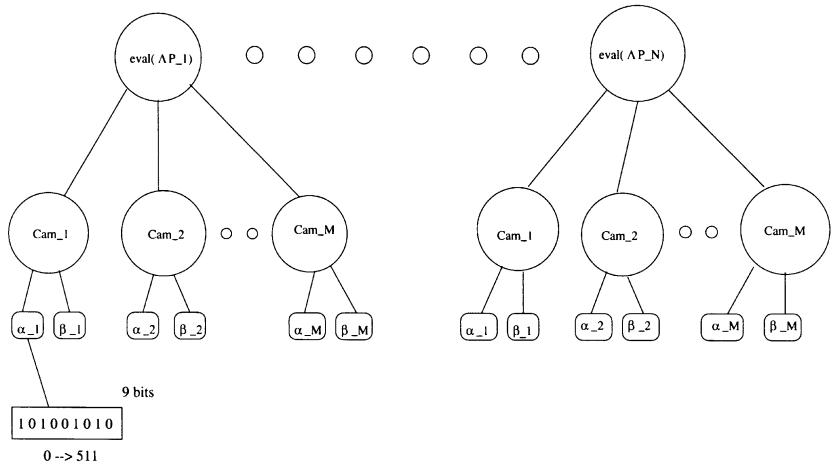


Fig. 5. The multi-cellular genetic algorithm is represented by a tree structure composed of a main node where the evaluation process is stored and several leaves correspond to each camera. All the cameras are codified in two parameters (α, β), which correspond to the cells of an artificial being.

2. The environment, $E \in \varepsilon$, which limits the structure, is represented here as the set of geometrical and optical constraints.
3. A measure μ_E of performance, i.e., the fitness of the structures for the environment, is represented here as the value $\mu(P)$.
4. The adaptive plan, $\Gamma \in \tau$, whereby the system's structure is modified to effect improvements. This is the genetic algorithm detailed later on.
5. The operator's set, $\Omega \in \omega$, which is used by the adaptive plan. This is represented here by the crossover and mutation operations.

These five main points are fundamental in establishing a strategy for the camera network design problem. This section describes a multi-cellular genetic algorithm (MGA) used to solve the camera position and orientation problem in order to obtain the minimal 3D error. The idea is to create a computational model to optimize a set of convergent camera networks using an evolutionary technique. This model lets us to implement the design as a problem of *optimization*. We propose a specialized implementation of the standard genetic algorithm approach, following the tree-based genetic programming representation [29,30], which we assume here as known and fixed,

see Fig. 5. This representation partitions the parameters into several distinct parts or "cells"; each one corresponding to one camera variable. Each camera evolves only with cameras of the same class. This partition allows the emergence of subpopulations of good individuals, as will be seen in the experiments. As an example, all the variables of camera 1 of a tree structure are combined among the variables of camera 1 of other tree structures, and so on for the other cameras. As a matter of fact, some of these cells are then concatenated to form one long binary string. Binary representation is preferred here to float due to the fact that *two points close to each other in the representation space are not close to each other in the solution space*. This allows a random search in the whole space. On the other hand, the probability of having a very large change in the value of the parameters when one has changed a single bit is rather small; only in some high-order bits. In this way, a robust structure is obtained which performs well with the changes of the environment constraints. Our algorithm does not measure the fitness of each cell individually. Instead, a global evaluation, before applying the geometrical and optical constraints, of a well-designed criterion using derivatives is implemented. A tree structure represents a camera network configuration that observes several parts of the object, where each part contains a group of targets. This camera network can be divided into several subnetworks; each one observing one group of targets. Then a genetic algorithm for each variable is executed, with just one general evaluation $\mu(P)$ among all the camera subnetworks. The fitness will then be the worst among these measures for a given object. Thus, we can see the algorithm as an idealized genetic algorithm IGA [31], where each partition is sampled independently and the best schema, set of bits, in each partition tends to be selected.

To keep the search space reasonably small, we use the viewing sphere model, see Fig. 12a, where the cameras move on a sphere looking in-wards towards a central point. This model provides convergent configurations which give an improved object measurement precision compared to other network configurations [32]. In this way, cameras move in the space to achieve the design goals. The direct relationship between the spherical coordinates of the viewing sphere and the components of imaging geometry allows the heuristic search of the genetic algorithm to be exploited. Consequently, the spherical coordinate system (α, β, γ) is well suited to the representation of the search space. All cameras are codified by the two parameters α and β, as the distance between the central point and the cameras r, is considered as a constant here. If the environment constraints such as the field of view and resolution are taken into account, the variable r will change. However, this does not change the final network configuration found by our system. Thus, the MGA is represented by a fixed tree structure composed of a main node where the evaluation process (the maximum value in the diagonal of ΛP) is stored, and several leaves correspond to each camera [33].

2.5.2. The multi-cellular genetic algorithm

The multi-cellular genetic algorithm then proceeds as follows:

1. An initial random population of N convergent networks that satisfy the environment constraints is chosen and is represented by (α_n, β_n), coded into a binary string representation.
2. Next, we evaluate each network, and store the corresponding maximum value of the diagonal ΛP_n for each tree structure. This corresponds to the fitness value which expresses how good the network is, compared with other solutions in the population $P(t)$.
3. Then, we select a population of "good" networks by *tournament selection* [34]: two networks are selected from $P(t)$ and are compared selecting the best individual according to its fitness, yielding the population $P(t+1)$.
4. From this population, we recombine the binary strings (α_n, β_n) for each camera using the following operations:
   - Crossover, with a probability[4] $P_c = 0.7$. This operation was implemented using *one-cut-point*[5]. Let the two parents be:

     $$\alpha_x = [\alpha_{x1}\ \alpha_{x2}\ \alpha_{x3}\ \alpha_{x4}\ \alpha_{x5}\ \alpha_{x6}\ \alpha_{x7}\ \alpha_{x8}\ \alpha_{x9}],$$

     $$\alpha_y = [\alpha_{y1}\ \alpha_{y2}\ \alpha_{y3}\ \alpha_{y4}\ \alpha_{y5}\ \alpha_{y6}\ \alpha_{y7}\ \alpha_{y8}\ \alpha_{y9}].$$

     If they are crossed after the random kth position $= 4$, the resulting offsprings are:

     $$\alpha'_x = [\alpha_{x1}\ \alpha_{x2}\ \alpha_{x3}\ \alpha_{x4}\ \alpha_{y5}\ \alpha_{y6}\ \alpha_{y7}\ \alpha_{y8}\ \alpha_{y9}],$$

     $$\alpha'_y = [\alpha_{y1}\ \alpha_{y2}\ \alpha_{y3}\ \alpha_{y4}\ \alpha_{x5}\ \alpha_{x6}\ \alpha_{x7}\ \alpha_{x8}\ \alpha_{x9}].$$

   - Mutation, with a probability $P_m = 0.005$. This operation alters one or more genes. Assume that $\alpha_{y5} = 1$ gene of the chromosome α'_x is selected for a mutation. Since the gene is 1, it would be flipped into 0.

     These operations yield a new population, which we copy into $P(t)$.
5. Steps 2–4 are repeated until the optimization criterion stabilizes.

Finally, this algorithm minimizes the maximum value in the diagonal of ΛP:

$$\textit{fitness} = \min_{i=1,\ldots,N} (\mu(P_i)). \tag{21}$$

Thereby, the camera placement M_i relative to the world coordinate frame is optimized. Geometrically, each ΛP_i

[4] For a discussion of the threshold values associated to P_c and P_m see Ref. [31].

[5] Due to the classification of the MGA, this operation works like a *multiple-cut-point*.

represents a hyperellipsoid, which changes its orientation and size as each sensor placement M_i does. Thus, an optimal placement solution is proposed, where the combined uncertainty of all points is minimal.

3. Experiments

We have run a series of experiments to test the validity of our approach. We present some results in Figs. 11 and 12, which show several configurations designed by EPOCA. The cameras are looking into a set of targets represented by their error ellipsoids aligned in one, two, or three planes, as well as over a complex object. These configurations are a product of our evolutionary system. In fact, within a stochastic optimization process we cannot make conclusions from just one trial. Actually, each configuration presented is the product of about 50 independent trials.

Fig. 12a presents a classical configuration used in photogrammetry to test the goodness of the mathematical model. Three cameras are placed over a central point using the viewing sphere model. The cameras are separated 120° from each other along the Y-axis. This configuration is used to study the relationship between the accuracy of the network and the angle of convergence. The same experiment is repeated considering several images taken by each camera. In this way, cameras are placed over the X–Z plane for different convergence angles, $\alpha = \{70, 60, \ldots, 30\}$ and for $2, 3, \ldots, 9$ images. Fig. 12c shows several curves representing the accuracy of each configuration [18,19,35]. Each curve is computed using the inverse of the error associated to the Z- and Y-axis. The values corresponding to the Y-axis are smaller. Since the X- and Z-axis are closer in magnitude we plot only the Z-axis. In this way, two curves are plotted. These curves correspond to the experimental results obtained by photogrammetrists in the last 20 years. If the requirement is for configuration with a good imaging geometry, the convergence angle must be around 45°, that is to say, a separation of 90° approximately. Indeed, all the curves cross at this value.

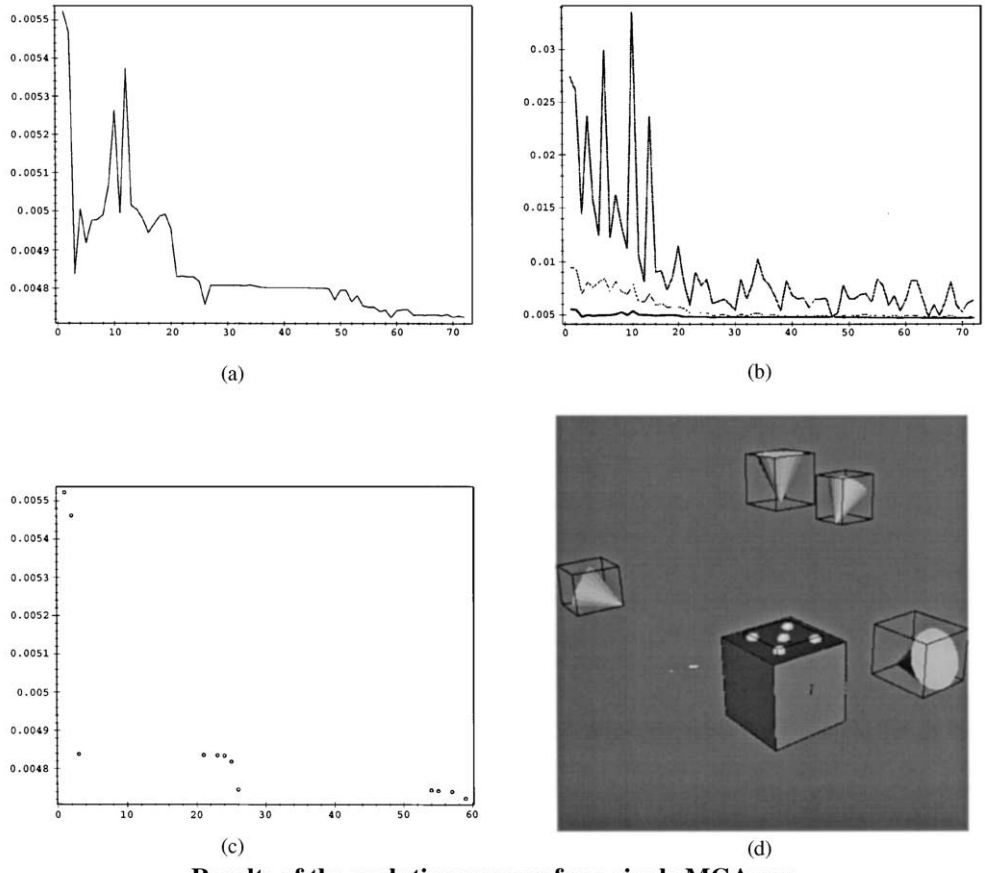


Fig. 6. The graphs show the statistics and convergence time of a single MGA run composed of 4 convergent cameras; (a) The plot of best fitness. (b) Maximal, average and best fitness. (c) The last best configuration. (d) The best configuration.

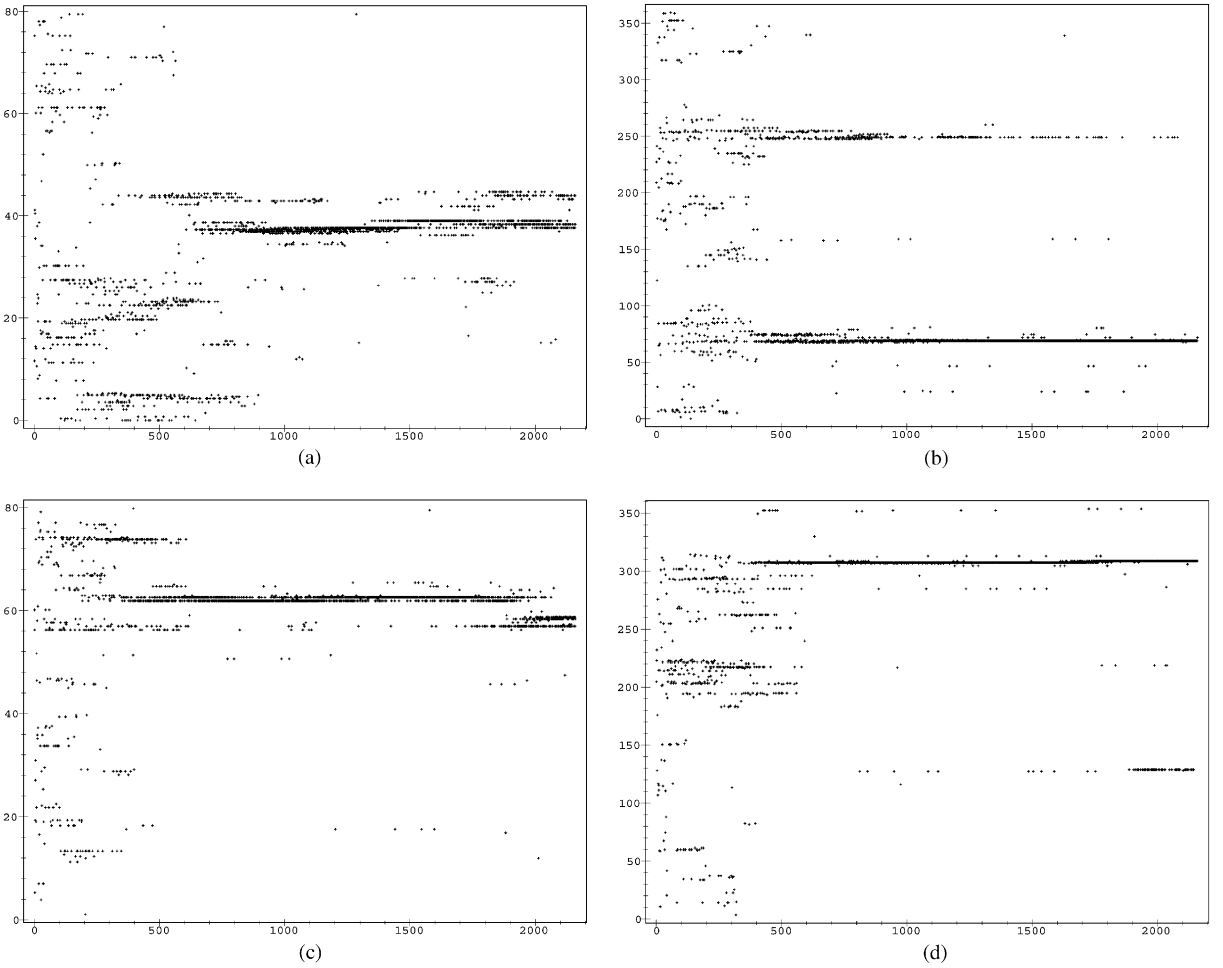


Fig. 7. The graphs show the evolution of cameras 1 and 2 represented by their variable positions (angles α and β). The Y-axis is in degrees and the X-axis shows the number of the considered sample. It is important to notice that each graph is divided into 72 generations, each one containing 30 samples, which represents 2160 samples; (a) Angle α of camera 1. (b) Angle β of camera 1. (c) Angle α of camera 2. (d) Angle β of camera 2.

Table 2
Final results of the four-camera network

Camera	Angle α	Angle β
1	39.02	68.90
2	62.92	308.67
3	0.00	176.48
4	64.33	184.21

As an example of a typical run, the graphs of Fig. 6 were produced with a population of 30 networks composed of four cameras which converged in no more than 150 generations. These graphs show the maximal, average and best fitness for one run of the algorithm. These results are similar to the four-camera network of Fig. 11b. This experiment took a total of 72 generations to converge, finding the best configuration with a fitness value $=4.7 \times 10^{-3}$ in generation 59. It required about 7 s of CPU time on an UltraSPARC 200 MHz. Fig. 6c shows the moment when a minimum appears along the run. Note that the evolution is terminated near generation 60. Table 2 presents the final results of the four-camera network configuration. Note the behaviour of the camera population as illustrated in Figs. 7 and 8. We see that after a random distribution, emergent coherent configurations appear. For instance, the α angle of camera 1, see Fig. 7a, converges towards two dominating values below and above 40°, while the β angle in the same camera is about 69°. The latter has a second

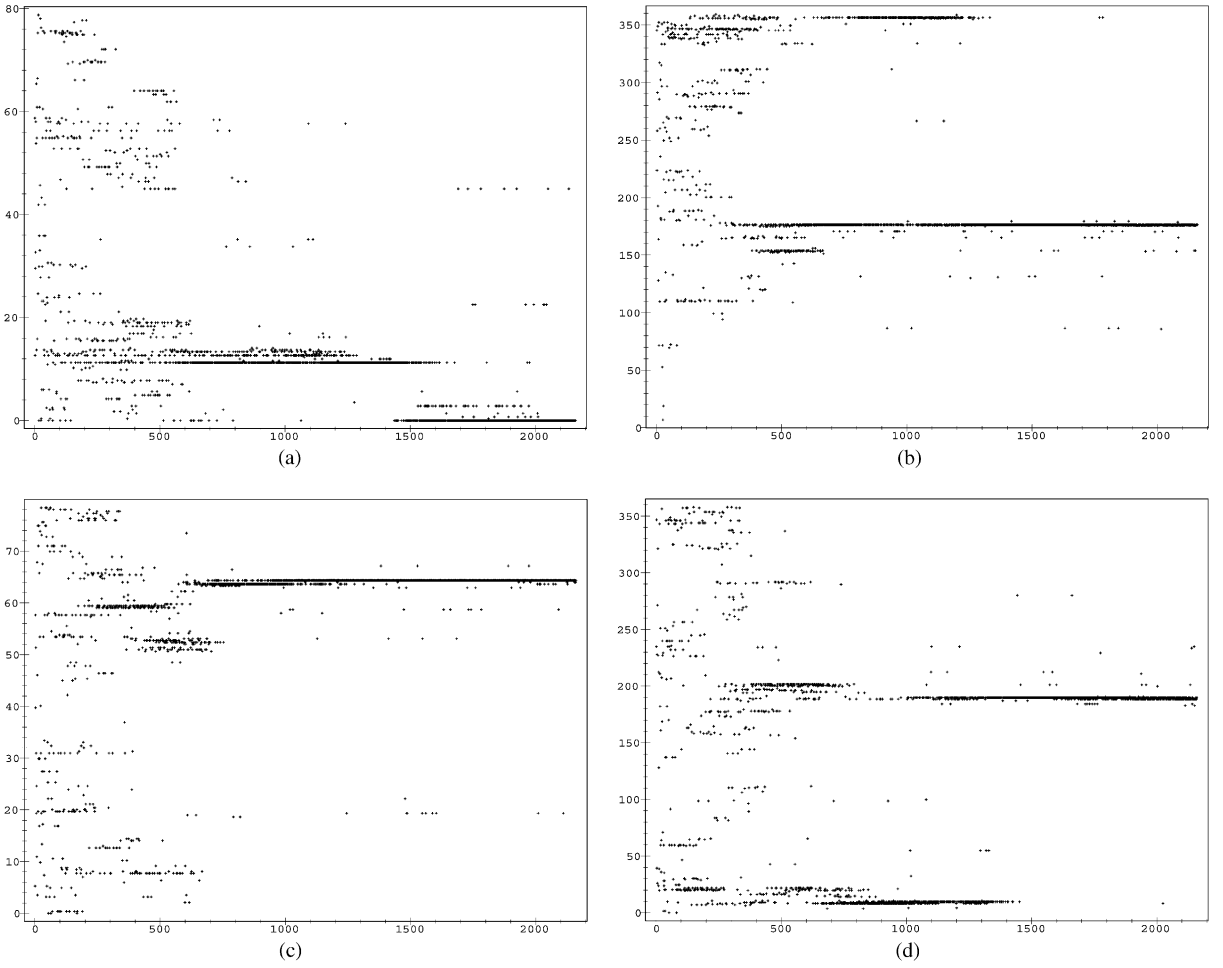


Fig. 8. The graphs show the evolution of cameras 3 and 4 represented by their variable positions (angles α and β). The Y-axis is in degrees and the X-axis shows the number of the considered sample. It is important to notice that each graph is divided into 72 generations, each one containing 30 samples, which represents 2160 samples; (a) Angle α of camera 3. (b) Angle β of camera 3. (c) Angle α of camera 4. (d) Angle β of camera 4.

minimum around 250°, see Fig. 9c, 180° after the first minimum. Other dominating configurations appear too, but disappear before half of the total iteration. Similar observation can be made on all other parameters.

If we analyze how $\mu(P)$ evolves with these parameters, we will observe the local minima of the dominating configurations. Fig. 9 displays the graph of $\mu(P)$, after convergence, for cameras 1 and 2 with respect to parameters α and β. It clearly shows the two minima for the β parameter of camera 1, while the landscape is quite flat for the α parameter. Similar observations can be made from the graphs of Fig. 10. Cameras 2 and 4 also present two local minima, in contrast to camera 3, that presents just one minimum. The α angle is about 0° (i.e., camera 3 is orthogonal to the surface plane); indeed, this is a singular configuration for which the parameter β of camera 3 can take any value. In general, the population size must be large enough to allow the MGA to find the good solutions. A small population (less than 10 networks) discourages premature convergence to suboptimal solutions. This is mainly due to the need for niches, which are landscapes from which the global or optimal solution is chosen. Consequently, if these zones are poorer, the chances to obtain a better design are reduced. On the other hand, a large population requires more evaluations per generation to achieve the same fitness values. The problem of convergence is usually related to the length of the string [36]. In our implementation, strings of 300 or more bits have been optimized owing to the tree classification. Other examples for two and three adjoining

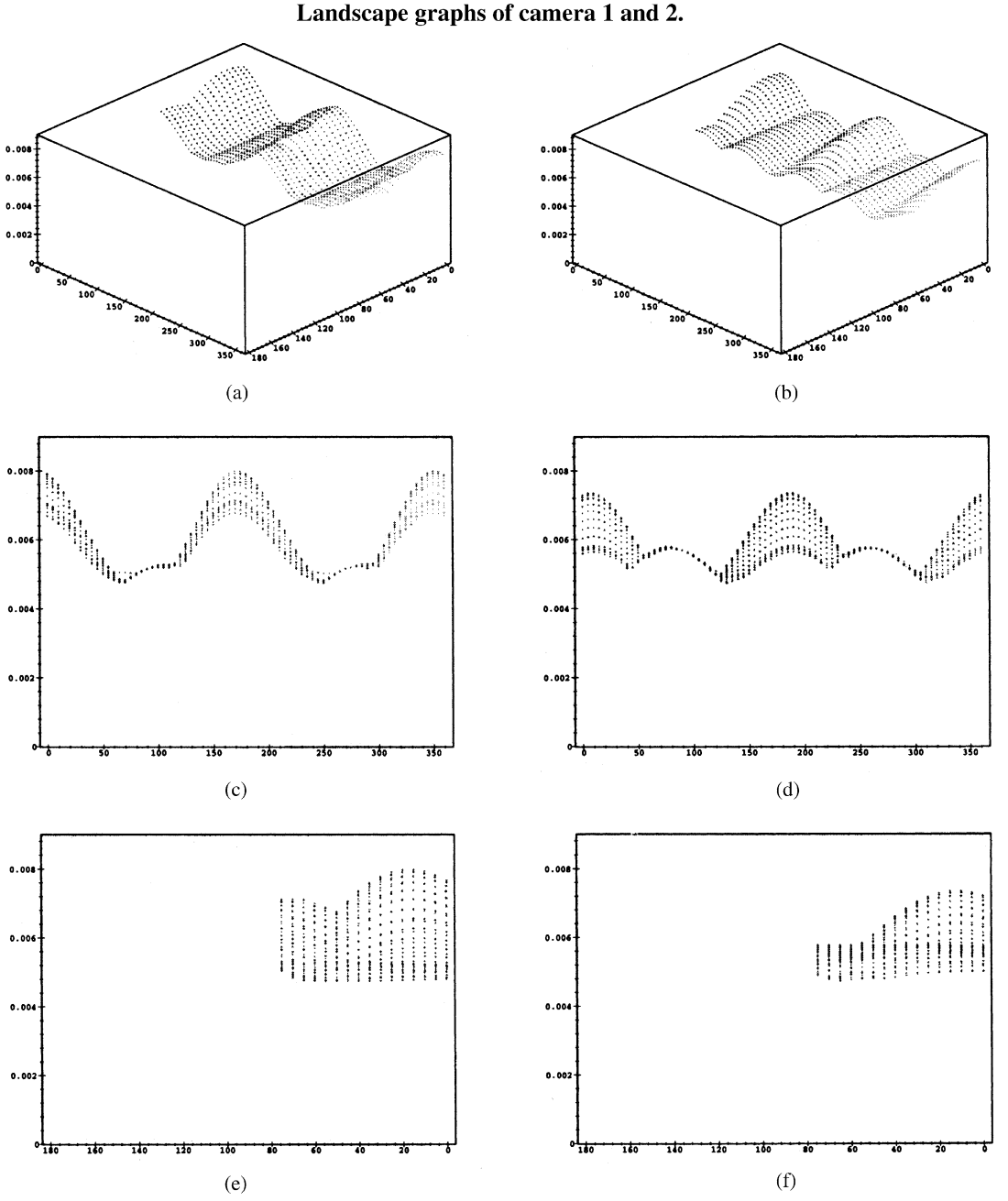


Fig. 9. The graphs on the left correspond to the landscape of camera 1, while the graphs on the right correspond to the landscape of camera 2; (a) Surface graph of camera 1. (b) Surface graph of camera 2. (c) Frontal view of graph a. (d) Frontal view of graph b. (e) Lateral view of graph a. (f) Lateral view of graph b.

planes are presented in Fig. 11. These examples show how the cameras are positioned at very precise zones of the space for which the overall uncertainty is minimal. Note that in the cases of three planes, the cameras do not observe all the targets at the same time.

Fig. 11b illustrates a solution with four cameras looking at a planar surface. This solution is not the standard one used by the experts: a photogrammetrist usually puts the four cameras at four corners of a cube whose center contains the targets to be measured. In fact, Fraser [1] has

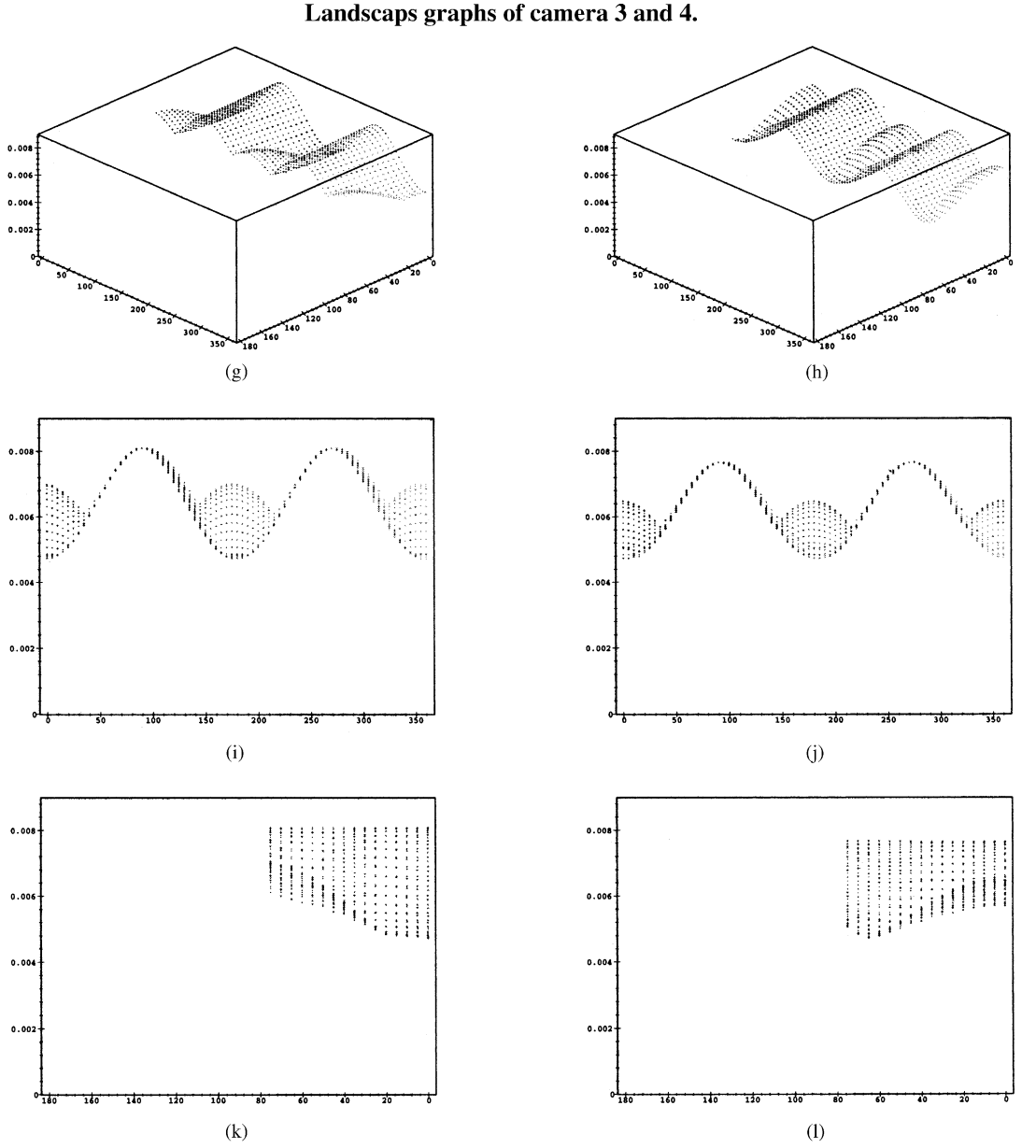


Fig. 10. The graphs on the left correspond to the landscape of camera 3, while the graphs on the right correspond to the landscape of camera 4; (g) Surface graph of camera 3. (h) Surface graph of camera 4. (i) Frontal view of graph g. (j) Frontal view of graph h. (k) Lateral view of graph g. (l) Lateral view of graph h.

already discussed our configuration; he noticed that this configuration is not atypical. Our experiments confirm Fraser's statement, hence the equivalence between both configurations, see Ref. [37]. Fig. 12b presents an interesting result. Cameras are placed over the same places of Fraser's configuration. This result corresponds to the *second-order design* (SOD) used by photogrammetrists. This operation consists in the acquisition of multiple exposures from each camera composing the network. The SOD operation is used normally after having selected a basic configuration. The multiple exposures thus yielded a good improvement in the mean standard error value over those obtained in the basic network. In the case of Fig. 12e, just 20 trials were necessary. This last result

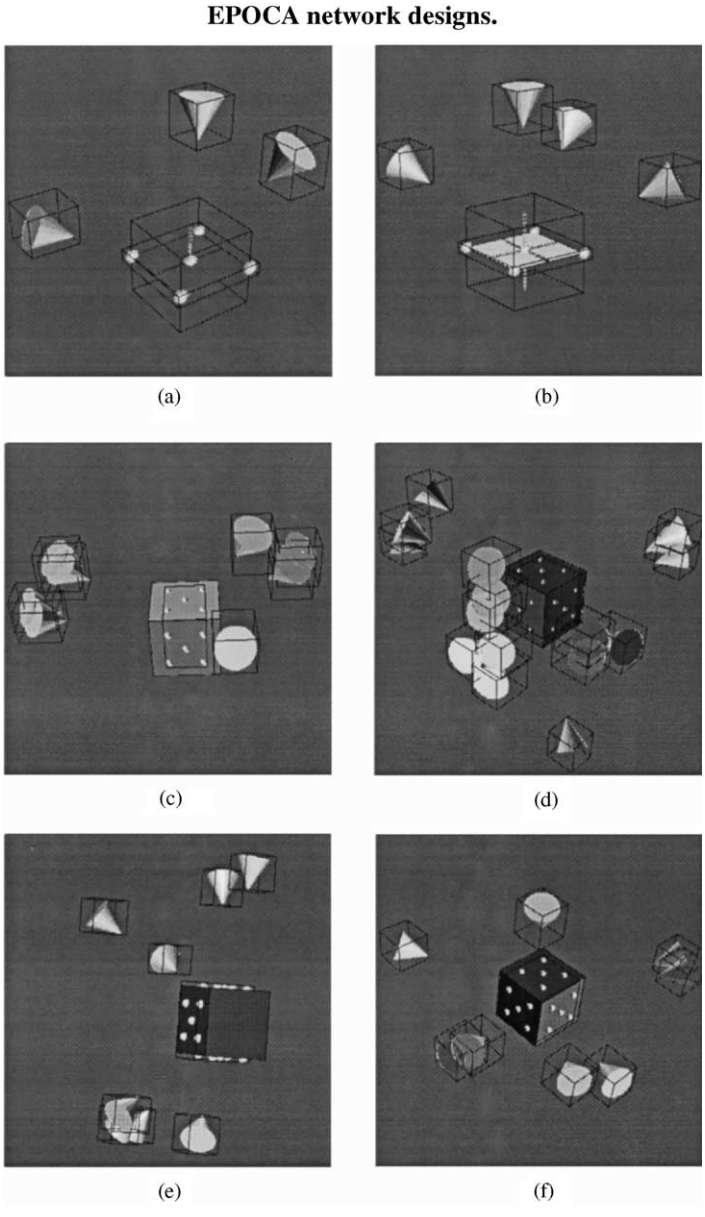


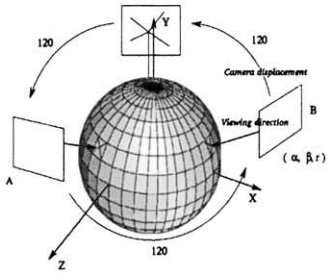
Fig. 11. Among the several designs proposed by EPOCA, we have identified design b) as the one used by Fraser [1], which is not atypical of an imaging geometry; (a) 3 cameras observing a plane. (b) 4 cameras over a plane. (c) 8 cameras over 2 adjoining planes. (d) 16 cameras over 3 adjoining faces. (e) 8 cameras observing 3 opposites faces. (f) 8 cameras over 3 adjoining faces.

shows how the object shape constraints the search space where cameras should be placed. All the examples presented correspond well to the generic network theory. In fact, results presented in this paper demonstrate how our system is able to propose solutions used by experts. Moreover, our system can contribute to finding solutions in the case of complex objects [38].

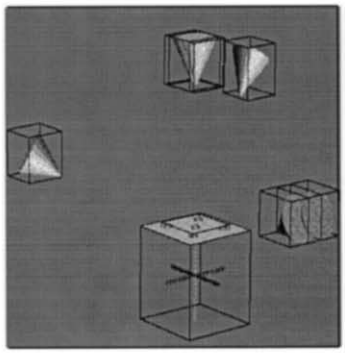
4. Discussion and perspectives

In this paper, we have presented a solution to the problem of optimal camera placement with the goal of achieving highly accurate 3D measurements in terms of an optimization design. The problem has been divided into two main parts. The first was devoted to an analytical

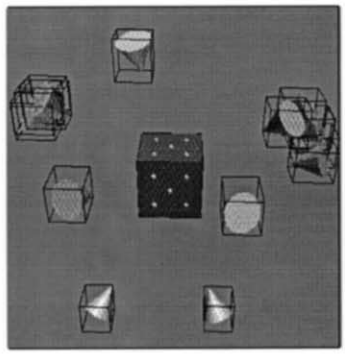
EPOCA network designs.



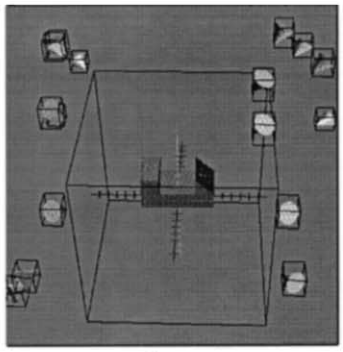
(a)



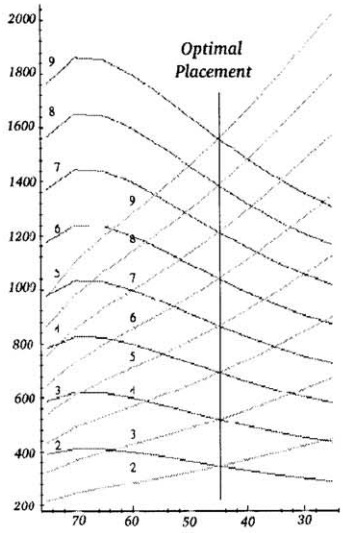
(b)



(d)



(e)



(c)

Examples of configurations designed by EPOCA.

Fig. 12. The curves plotted in Fig. 12c show the optimal placement corresponding to the imaging geometry of Fig. 12a. Other examples of configurations reported in the literature were reproduced by EPOCA, see Figs. 12b and 12d. Moreover, EPOCA can be used in the case of complex objects, as can be appreciated from Fig. 12e; (a) 3 camera configuration to study the imaging geometry. (b) Configuration similar to Figure 6d. (a) Camera distribution is evaluated at different angles of convergence and number of images. These curves show that at 45 degrees an optimal placement is achieved in all cases. (d) Configuration similar to one proposed by Mason and Grün [15]. (e) 16 cameras over a complexs object.

error model from which a criterion was derived. The criterion we have chosen is the *maximum element in the diagonal* extracted from the covariance matrix, which is based on the error propagation phenomenon. This criterion was chosen in order to speed up the MGA process.

The second part was devoted to a global optimization method, which has minimized the criterion. Due to the

occlusion of points, caused by the different constraints, the camera network design problem presents discontinuities, which leads to combinatorial aspects in the optimization process. These constraints are naturally incorporated into the genetic algorithm methodology. The system is a dynamic process where the global state *emerges* by coevolution through the behavior and interaction of the "cells" which are related by coadaptation. It is necessary to mention that the behavior of these cells is influenced by the global state. An extrinsic parallelism, together with the intrinsic parallelism, is executed. A phenomenon of *niche* is present in each camera representation. These niches evolve with respect to the others mainly due to the rules of the system. Our work confirms results obtained by other researchers [39]: genetic algorithms are able to solve such kind of optimization problems with high combinatorial aspects efficiently.

Our EPOCA system successfully produces two and three camera network designs similar to those used by photogrammetrists. In the case of four cameras, a non-standard design was proposed which would give similar results to those obtained from the more classical networks. Moreover, the system can design networks for several adjoining planes and complex objects. All the configurations are good in terms of the camera distribution and ray inclination. Furthermore, all our results agree well with empirical results obtained by photogrammetrists. This work is to be considered as an additional step towards an automated camera network design. The strength of our approach is its generality. Consideration of alternative optimization processes working on a non-uniform criterion could easily extend this work. However, several simplifying assumptions made here will have to be explored in the future. We assume that the external and internal parameters of the camera are perfectly known. In fact, the user has good estimates for them, but if high accuracy were needed, the bundle adjustment would have to refine its estimates together with the 3D-measurement estimation. This complicates the computation of the criterion at each evaluation step in the optimization. This aspect will increase the optimization time as well as the non-linearity, and the open questions that arise are: Will such a system converge? How long will it take? While time might not be a major theoretical issue, it is a practical one, as a huge scale factor could multiply the computation time. These aspects are both related to the computational complexity of the optimization; our goal for future research is to explore them.

Acknowledgements

This research was funded by contract 35267-A from CONACyT. This article is based on the first author's doctoral dissertation. We are also grateful to Dr. Scott Mason and Dr. Marc Schoenauer for their helpful comments. Figs. 1, 6d and 11 were generated with software written at the Geometry Center.

References

[1] C.S. Fraser, Optimization of precision in close-range photogrammetry, Photogrammetric Eng. Remote Sensing 53 (5) (1982) 487–493.
[2] H.A. Beyer, Geometric and radiometric analysis of a CCD-camera based photogrammetric close-range system, Ph.D. Thesis, ETH-Zurich, May 1992.
[3] S. Sakane, T. Sato, Automatic planning of light source and camera placement for an active photometric stereo system, Proceedings of IEEE International Conference on Robotics an Automation, Sacramento, California, USA, IEEE Robotics and Automation Society, April 1991, pp. 1080–1087.
[4] S. Sakane, R. Niepold, T. Sato, T. Shirai, Illumination setup planning for hand-eye system based on an environmental model, Adv. Robot. 6 (4) (1992) 461–482.
[5] S. Yi, R.M. Haralick, L.G. Shapiro, Optimal sensor and light source positioning for machine vision, Comput. Vision Image Understanding 61 (1) (1995) 122–137.
[6] C.K. Cowan, P.D. Kovesi, Automatic sensor placement from vision task requirements, IEEE Trans. Pattern Anal. Mach. Intell. 10 (3) (1988) 407–416.
[7] K.A. Tarabanis, R.Y. Tsai, Computing view points that satisfy optical constraints, Proceedings of the Conference on Computer Vision and Pattern Recognition, Maui, Hawaii, USA, 1991, pp. 152–158.
[8] S. Abrams, P.K. Allen, K.A. Tarabanis, Dynamic sensor planning, IEEE International Conference on Robotics and Automation, Atlanta, GA, May 1993.
[9] C.K. Cowan, B. Modayur, J. DeCurtins, Automatic light-source placement for detecting object features, Intelligent Robots and Computer Vision XI, Vol. 1826, SPIE-Society of Photo-Optical Instrumentation, Boston, 1992, pp. 397–408.
[10] R. Pito, A solution to the next best view problem for automated surface acquisition, IEEE Trans. Pattern Anal. Mach. Intell. 21 (10) (1999) 1016–1030.
[11] G.H. Tarbox, S.N. Gottschlich, Planning for complete sensor coverage in inspection, Technical Report TR-CAT-93-4, NYS Center for Advanced Technology in Automation and Robotics, 1993.
[12] C.K. Cowan, A. Bergman, D. Nitzan, Automatic placement of vision sensors, Proceedings 1990 NSF Manufacturing System Research Conference, Vol. 1, 1990, pp. 389–395.
[13] D. Fritsch, F. Crosilla, First order design strategies for industrial photogrammetry, Close-Range Photogrammetry Meets Machine Vision, Vol. 1395, SPIE-Society of Photo-Optical Instrumentation Engineers, 1990, pp. 432–438.
[14] S. Mason, Heuristic reasoning strategy for automated sensor placement, Photogramm. Eng. Remote Sensing 63 (9) (1997) 1093–1102.
[15] S. Mason, A, Grün, Automatic sensor placement for accurate dimensional inspection, Comput. Vision Image Understanding 61 (3) (1995) 454–467.

[16] G. Olague, R. Mohr, Optimal camera placement to obtain accurate 3d point positions, 14th International Conference on Pattern Recognition, Vol. I, Brisbane, Australia, APRS and IAPR, August 16–20 1998, pp. 8–10.
[17] D. Goldberg, Genetic Algorithms in Search Optimization and Machine Learning, Addison-Wesley, Reading, MA, 1989.
[18] D.C. Brown, Application of close-range photogrammetry to measurements of structures in orbit, Technical Report 80-012, Geodetic Services Incorporated, Melbourne Florida, Vol. 1, September 15 1980, pp. 131.
[19] C.S. Fraser, Limiting error propagation in network design, Photogramm. Eng. Remote Sensing 48 (4) (1987) 561–570.
[20] O. Faugeras, Three-Dimensional Computer Vision A Geometric Point of View, MIT Press, Cambridge, MA, 1996.
[21] R. Mohr, Projective geometry and computer vision, in: C.H. Chen, L.F. Pau, S.P. Wang (Eds.), Handbook of Pattern Recognition and Computer Vision, World Scientific Pub., Singapore, 1993.
[22] L. Morin, Quelques contributions des invariants projectifs a la visión par ordinateur, Ph.D. Thesis, Institut National Polytechnique de Grenoble, Janvier, 1993.
[23] A.W. Gruen, Adaptive least squares correlation: a powerful image matching technique, S. Afr. J. Photogrammetry Remote Sensing Cartography 14 (3) (1985) 175–187.
[24] W.H. Press, B.P. Flannery, S.A. Teukolsky, W.T. Vetterling, Numerical Recipes in C, 2nd Edition, Cambridge University Press, Cambridge, 1988.
[25] W. Förstner, A metric for comparing symmetric positive definite matrices, Institute of Photogrammetry, August 1995.
[26] J.H. Holland, Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence, 2nd Edition, MIT Press, Cambridge, MA, 1992 (first appear in 1975).
[27] S. Mason, Conceptual model of the convergent multistation network configuration task, Photogrammetric Record 15 (86) (1995) 227–299.
[28] G. Olague, Design and simulation of photogrammetric networks using genetic algorithms, ASPRS 2000 Annual Conference Proceedings, Washington DC, USA, American Society for Photogrammetry & Remote Sensing, p. 12, Copyright © May 22–26, 2000.
[29] J.R. Koza, Genetic Programming, On the Programming of Computers by Means of Natural Selection, The MIT Press, Cambridge, MA, USA, 1992.
[30] K.E. Kinnear, Advances in Genetic Programming, The MIT Press, Cambridge, MA, USA, 1994.
[31] M. Mitchell, An Introduction to Genetic Algorithms, A Bradford Book, 1st Edition, Massachusetts Institute of Technology, 1996.
[32] C.S. Fraser, Network design, in: K.B. Atkinson (Ed.), Close Range Photogrammetry and Machine Vision, 1st Edition, Whittles Publishing, Caithness, Scotland, 1996, pp. 256–281.
[33] G. Olague, Autonomous photogrammetric network design using genetic algorithms, in: E.J.W. Boers, et al., (Eds.), Applications of Evolutionary Computing, Lecture Notes in Computer Science, Vol. 2037, Springer, Lake Como, Italy, April 18 2001, p. 10.
[34] T.L. Blickle, A mathematical analysis of tournament selection, Proceedings of the Sixth International Conference on GA, Univ. of Pittsburgh, July 1995, pp. 9–16.
[35] W.J. Oh, S.H. Han, S.H. Song, S.H. Bae, H.S. Lee, A study on the development of semi-metric camera for very close-range photogrammetry, International Archives of Photogrammetry and Remote Sensing, Vol. XXXI, Part B5, Vienna, 1996, pp. 426–431.
[36] M. Mitchell, S. Forrest, J.H. Holland, The royal road for genetic algorithms: Fitness landscapes and ga performance, Proceedings of the Ist European Conference on Artificial Life, MIT Press, Cambridge, MA, December, 11–13 1992, pp. 245–254.
[37] S.O. Mason, Expert system-based design of photogrammetric networks, Ph.D. Thesis, Institut für Geodäsie und Photogrammetrie, Swiss Federal Institute of Technology (ETH) CH-8093 Zürich, May 1994.
[38] G. Olague, Planification du placement de caméras pour des measures 3D de précision, Ph.D. Thesis, Institut National Polytechnique de Grenoble, ftp://ftp.imag.fr/pub/Mediatheque.IMAG/theses/98-Olague.Gustavo/notice-francais.html, Octobre 1998.
[39] P. Bessiére, J.M. Ahuactzin, E. Talbi, E. Mazer, The ariadnés clew algorithm: global planning with local methods, in: Latombe Goldberg, Halperin, Wilson (Eds.), Algorithm Foundations of Robotics, A.K. Peters, Boston, 1995.

About the Author—GUSTAVO OLAGUE holds a Bachelor's degree (Honors) in Electronics Engineering and a Master's degree in Computer Science from the Instituto Tecnológico de Chihuahua, México. He received the "Diplôme de Doctorat en Imagerie, Vision et Robotique" (Ph.D.) from Institut National Polytechnique de Grenoble, France. He is presently an Assistant Professor of Computer Science at the Centro de Investigación Científica y de Estudios Superiores de Ensenada, México. Olague's research focuses on the principles of computational intelligence applied to close-range photogrammetry and computer vision. He is a member of the ASPRS, IEEE and IEEE Computer Society.

About the Author—ROGER MOHR is a professor at the Institut National Polytechnique de Grenoble, "Directeur de Recherche au CNRS" and the head of the MOVI research group. His research interests include geometry for 3D computer vision, 3D perception, and image indexing and retrival.